

The vanishing ideal of a finite set of points with multiplicity structures

Na Lei, Xiaopeng Zheng, Yuxue Ren

School of Mathematics, Jilin University

Abstract

Given a finite set of arbitrarily distributed points in affine space with arbitrary multiplicity structures, we present an algorithm to compute the reduced Gröbner basis of the vanishing ideal under the lexicographic ordering. Our method discloses the essential geometric connection between the relative position of the points with multiplicity structures and the quotient basis of the vanishing ideal, so we will explicitly know the set of leading terms of elements of I . We split the problem into several smaller ones which can be solved by induction over variables and then use our new algorithm for intersection of ideals to compute the result of the original problem. The new algorithm for intersection of ideals is mainly based on the Extended Euclidean Algorithm.

Keywords: vanishing ideal, points with multiplicity structures, reduced Gröbner basis, intersection of ideals.

1. Introduction

To describe the problem, first we give the definitions below.

Definition 1: $D \subseteq \mathbb{N}_0^n$ is called a lower set as long as $\forall d \in D$ if $d_i \neq 0$, $d - e_i$ lies in D where $e_i = (0, \dots, 0, 1, 0, \dots, 0)$ with the 1 situated at the i -th position ($1 \leq i \leq n$). For a lower set D , we define its limiting set $E(D)$ to be the set of all $\beta \in \mathbb{N}_0^n - D$ such that whenever $\beta_i \neq 0$, then $\beta - e_i \in D$.

As showed in Fig.1 below, there are three lower sets and their limiting sets. The elements of the lower sets are marked by solid circles and the elements of the limiting sets are marked by blank circles.

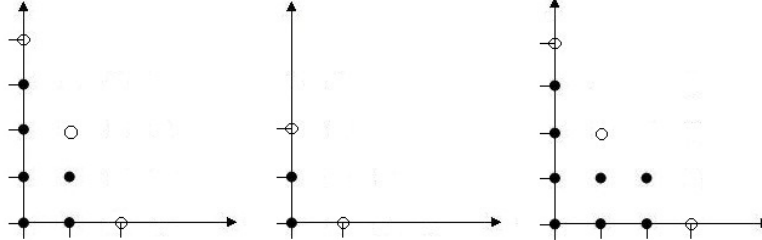


Fig.1: Illustration of three lower sets and their limiting sets.

Let k be a field and p be a point in the affine space k^n , i.e. $p = (p_1, \dots, p_n) \in k^n$. Let $k[X]$ be the polynomial ring over k , where we write $X = (X_1, X_2, \dots, X_n)$ for brevity's sake.

Definition 2: $\langle p, D \rangle$ represents a point p with multiplicity structure D , where p is a point in affine space k^n and D is a lower set. $\sharp D$ is called the multiplicity of point p (here we use the definition in [3]). For each $d = (d_1, \dots, d_n) \in D$, we define a corresponding functional

$$L(f) = \frac{\partial^{d_1+\dots+d_n}}{\partial x_1^{d_1} \dots \partial x_n^{d_n}} f(p).$$

Hence for any given finite set of points with multiplicity structures $H = \{\langle p_1, D_1 \rangle, \dots, \langle p_t, D_t \rangle\}$, we can define m functionals where $m \triangleq \sharp D_1 + \dots + \sharp D_t$. Our aim is to find the reduced Gröbner basis of the vanishing ideal $I(H) = \{f \in k[X]; L_i(f) = 0, i = 1, \dots, m\}$ under the lexicographic ordering with $X_1 \succ X_2 \succ \dots \succ X_n$.

There exists an algorithm that provides a complete solution to this problem in [4]. However, our answer for the special case of lexicographical ordering will be in a way more transparent than the one above. The ideas are summed-up as follows:

- Construct the reduced Gröbner basis of $I(H)$ and get the quotient basis $D(H)$ by induction over variables.
- Get the quotient basis $D(H)$ purely according to the geometric distribution of the points with multiplicity structures.
- Split the original problem into smaller ones which can be converted into 1 dimension lower problems and hence can be solved by induction over variables.
- Compute the intersection of the ideals of the smaller problems by using Extended Euclidean Algorithm.

There are several publications which have a strong connection to the work presented here. Paper [5] give a computationally efficient algorithm to get the quotient basis of the vanishing ideal over a set of points with no multiplicity structures and the authors introduce the lex game to describe the problem. Paper [6] offers a purely combinatorial algorithm to obtain the linear basis of the quotient algebra which can handle the set of points with multiplicity structures but it does not give the Gröbner basis. For a finite set of points with multiplicity structures, our algorithm obtains a lower set by induction over variables and constructs the reduced Gröbner bases at the same time. It is only by constructing Gröbner basis we can prove that the lower set is the quotient basis.

One important feature of our method is the clear geometric interpretation, so in Section 2 an example together with some auxiliary pictures will be given in the first place to demonstrate this kind of feature which can make the algorithms and conclusions in this paper easier understood for us. In Section 3 and 4, some definitions and notions are given. Section 5 and 6 are devoted to our main algorithms of computing the reduced Gröbner basis and the quotient basis together with the proofs. In Section 7 we demonstrate the algorithm to compute the intersection of two ideals and some applications.

2. Example

First we give two different forms to represent the set of points H with multiplicity structures.

For easier description, we introduce the matrix form which consists of two matrices $\langle \mathcal{P} = (p_{i,j})_{m \times n}, \mathcal{D} = (d_{i,j})_{m \times n} \rangle$ with $\mathcal{P}_i, \mathcal{D}_i$ denoting the i -th row vectors of \mathcal{P} and \mathcal{D} respectively. Each pair $\{\mathcal{P}_i, \mathcal{D}_i\}$ ($1 \leq i \leq m$) defines a functional in the following way.

$$L_i(f) = \frac{\partial^{d_{i,1} + \dots + d_{i,n}}}{\partial x_1^{d_{i,1}} \dots \partial x_n^{d_{i,n}}} f|_{x_1=p_{i,1}, \dots, x_n=p_{i,n}}.$$

And the functional set defined above is the same with that defined by H in Section 1.

For example, given a set of three points with their multiplicity structures $\{\langle p_1, D_1 \rangle, \langle p_2, D_2 \rangle, \langle p_3, D_3 \rangle\}$, where $p_1 = (1, 1), p_2 = (2, 1), p_3 = (0, 2), D_1 = \{(0, 0), (0, 1), (1, 0)\}, D_2 = \{(0, 0), (0, 1), (1, 0), (1, 1)\}, D_3 = \{(0, 0), (1, 0)\}$, the matrix form is like the follows.

$$\mathcal{P} = \begin{pmatrix} 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 2 & 1 \\ 2 & 1 \\ 2 & 1 \\ 2 & 1 \\ 0 & 2 \\ 0 & 2 \end{pmatrix}, \mathcal{D} = \begin{pmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \\ 1 & 1 \\ 0 & 0 \\ 1 & 0 \end{pmatrix}.$$

For intuition's sake, we also represent the points with multiplicity structures in a more intuitive way as showed in the left picture of Fig.2 where each lower set which represents the multiplicity structure of the corresponding point p is also put in the affine space with the zero element $(0,0)$ situated at p . This intuitive representing form is the basis of the geometric interpretation of our algorithm.

We take the example above to show how our method works and what the geometric interpretation of our algorithm is like:

Step 1: Define mapping $\pi : H \mapsto k$ such that $\langle p = (p_1, \dots, p_n), D \rangle \in H$ is mapped to $p_n \in k$. So $H = \{\langle p_1, D_1 \rangle, \langle p_2, D_2 \rangle, \langle p_3, D_3 \rangle\}$ consists of two π -fibres: $H_1 = \{\langle p_1, D_1 \rangle, \langle p_2, D_2 \rangle\}$ and $H_2 = \{\langle p_3, D_3 \rangle\}$ as showed in the middle and the right pictures in Fig.2. Each fibre defines a new problem, so we split the original problem defined by H into two small ones defined by H_1 and H_2 respectively.

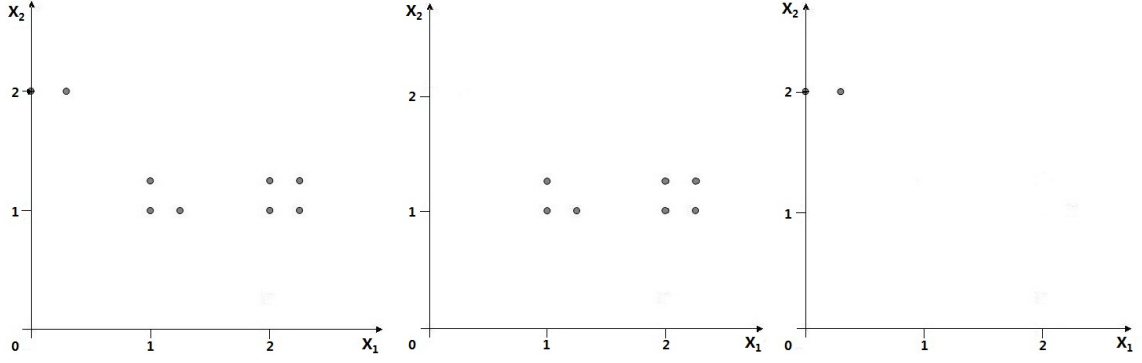


Fig. 2: The left picture represents H . The middle one is for H_1 and the right one for H_2 .

Step 2: Solve the small problems. Take the problem defined by H_1 for example.

First, it's easy to write down one element of $I(H_1)$:

$$f_1 = (X_2 - 1)(X_2 - 1) = (X_2 - 1)^2 \in I(H_1).$$

The geometry interpretation is: we draw two lines sharing the same equation of $X_2 - 1 = 0$ to cover all the points as illustrated in the left picture in Fig.3 and the corresponding polynomial is f_1 .

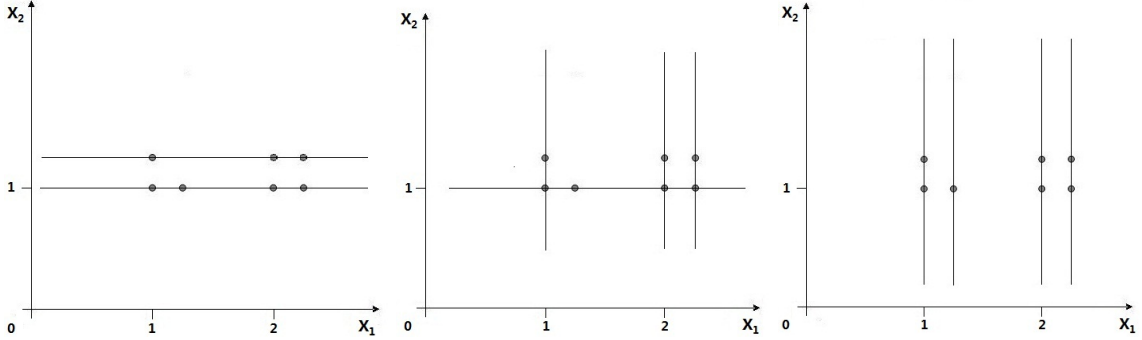


Fig. 3: Three ways to draw lines to cover the points.

According to the middle and the right pictures in Fig.2, we can write down another two polynomials in $I(H_1)$:

$$f_2 = (X_2 - 1)(X_1 - 1)(X_1 - 2)^2 \quad \text{and} \quad f_3 = (X_1 - 1)^2(X_1 - 2)^2.$$

It can be checked that $G_1 = \{f_1, f_2, f_3\}$ is the reduced Gröbner basis of $I(H_1)$, and the quotient basis is $\{1, X_1, X_2, X_1X_2, X_1^2, X_2X_1^2, X_1^3\}$. In the following, we don't distinguish explicitly an n -variable monomial $X_1^{d_1}X_2^{d_2}\dots X_n^{d_n}$ with the element (d_1, d_2, \dots, d_n) in \mathbb{N}_0^n , and we denote the quotient basis of $I(H)$ by $D(H)$. Hence $D(H_1)$ can be written as a subset of \mathbb{N}_0^n : $\{(0,0), (1,0), (0,1), (1,1), (2,0), (2,1), (3,0)\}$, i.e. a lower set, denoted by D' .

In fact we can get the lower set in a more direct way by pushing the points with multiplicity structures leftward which is illustrated in the picture below (lower set D' is positioned in the right part of the picture with the $(0,0)$ element situated at point $(0,1)$). The elements of the lower set D' in the right picture in Fig.4 are marked by solid circles. The blank circles constitute the limiting set $E(D')$ and they are the leading terms of the reduced Gröbner basis $\{f_1, f_2, f_3\}$.

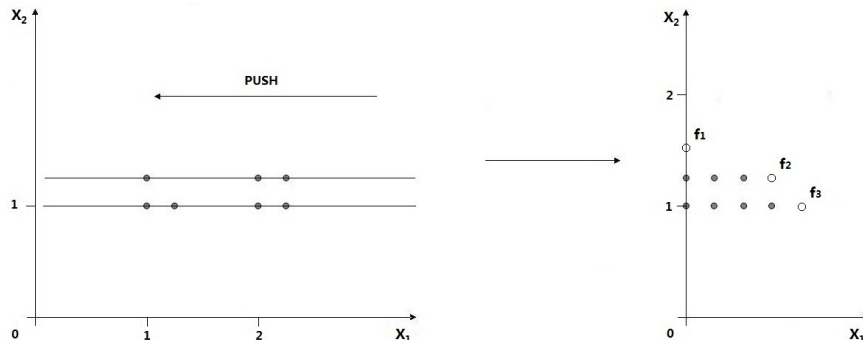


Fig.4: Push the points leftward to get a lower set.

In the same way, we can get the Gröbner basis $G_2 = \{h_1, h_2\}$ and a lower set D'' for the problem defined by H_2 , where $h_1 = (X_2 - 2), h_2 = X_1^2, D'' = \{(0, 0), (1, 0)\}$.

Step 3: Compute the intersection of the ideals $I(H_1)$ and $I(H_2)$ to get the result for the problem defined by H .

First, we construct a new lower set D based on D', D'' in an intuitive way: let the solid circles fall down and the elements of D'' rest on the elements of D' to form a new lower set D which is showed in the right part of Fig.5 and the blank circles represent the elements of the limiting set $E(D)$.

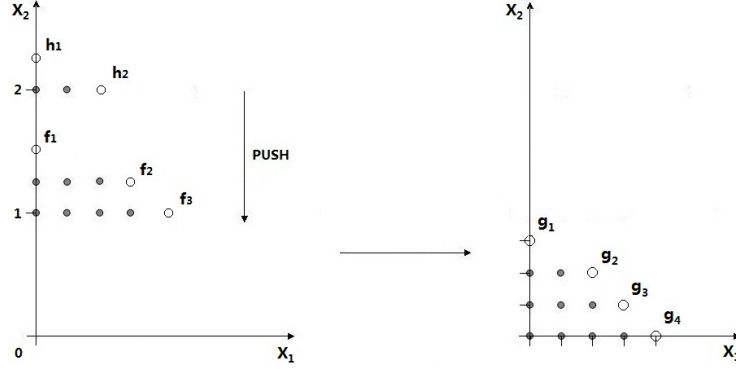


Fig. 5: Get the lower set D based on D' and D'' .

Then we need to find $\sharp E(D)$ polynomials vanishing on H with leading terms being the elements of $E(D)$. Take $X_1^3 X_2 \in E(D)$ for example to show the general way we do it.

We need two polynomials which vanish on H_1 and H_2 respectively, and their leading terms both have the same degrees of X_1 with that of the desired monomial $X_1^3 X_2$ and both have the minimal degrees of X_2 . It's easy to notice that f_2 and $X_1 \cdot h_2$ satisfy the requirement and then we multiply f_2 and $X_1 \cdot h_2$ with h_1, f_1 respectively which are all univariate polynomials of X_2 to get two polynomials q_1, q_2 which both vanish on H .

$$q_1 = f_2 \cdot h_1 = (X_2 - 1)(X_1 - 1)(X_1 - 2)^2(X_2 - 2),$$

$$q_2 = X_1 \cdot h_2 \cdot f_1 = X_1^3(X_2 - 1)^2.$$

Next try to find two univariate polynomials of X_2 : r_1, r_2 such that $q_1 \cdot r_1 + q_2 \cdot r_2$ vanishes on H (which is apparently true already) and has the desired leading term $X_1^3 X_2$.

To settle the leading term issue, write q_1, q_2 as univariate polynomials of X_1 . $q_1 = (X_2 - 2)(X_2 - 1)X_1^3 - (5X_2^2 - 15X_2 + 10)X_1^2 + (8X_2^2 - 24X_2 + 16)X_1 - 4X_2^2 + 12X_2 - 8, q_2 = (X_2 - 1)^2 X_1^3$. Because $X_2 < X_1$ and the highest degrees of X_1 of the leading terms of q_1, q_2 are both 3, we know that as long as the leading term of $(X_2 - 2)(X_2 - 1)X_1^3 \cdot r_1 + (X_2 - 1)^2 X_1^3 \cdot r_2$ is $X_1^3 X_2$, the leading term of $q_1 \cdot r_1 + q_2 \cdot r_2$ is also $X_1^3 X_2$.

$$(X_2 - 2)(X_2 - 1)X_1^3 \cdot r_1 + (X_2 - 1)^2 X_1^3 \cdot r_2$$

$$= X_1^3(X_2 - 1)((X_2 - 2) \cdot r_1 + (X_2 - 1) \cdot r_2)$$

Obviously if and only if $(X_2 - 2) \cdot r_1 + (X_2 - 1) \cdot r_2 = 1$ we can keep the leading term of $q_1 \cdot r_1 + q_2 \cdot r_2$ to be $X_1^3 X_2$. In this case $r_1 = -1$ and $r_2 = 1$ will be just perfect. In our algorithm we use Extended Euclid Algorithm to compute r_1, r_2 .

Finally we obtain

$$g_3 = q_1 \cdot r_1 + q_2 \cdot r_2 = (X_2 - 1)X_1^3 + (5X_2^2 - 15X_2 + 10)X_1^2 - (8X_2^2 - 24X_2 + 16)X_1 + 4X_2^2 - 12X_2 + 8$$

which vanishes on H and has $X_1^3 X_2$ as its leading term.

In the same way, we can get $g_1 = (X_2 - 1)^2(X_2 - 2)$ for X_2^3 , $g_2 = (X_2 - 1)^2 X_1^2$ for $X_1^2 X_2^2$ and $g_4 = X_1^4 + 6(X_2^2 - 2X_2)X_1^3 - 13(X_2^2 - 2X_2)X_1^2 + 12(X_2^2 - 2X_2)X_1 - 4(X_2^2 - 2X_2)$ for X_1^4 . In fact we need to compute g_1, g_2, g_3 and g_4 in turn according to the lexicographic order because we need reduce g_2 by g_1 , reduce g_3 by g_2 and g_1 , and reduce g_4 by g_1, g_2 and g_3 .

The reduced polynomial set can be proved in Section 6 to be the reduced Gröbner basis of the intersection of two ideals which is exactly the vanishing ideal over H , and D is the quotient basis.

3. Notions

First, we define the following mappings.

$$\begin{aligned} \text{proj} : \mathbb{N}_0^n &\longrightarrow k \\ (d_1, \dots, d_n) &\longrightarrow d_n. \\ \widehat{\text{proj}} : \mathbb{N}_0^n &\longrightarrow \mathbb{N}_0^{n-1} \\ (d_1, \dots, d_n) &\longrightarrow (d_1, \dots, d_{n-1}). \\ \text{embed}_c : \mathbb{N}_0^{n-1} &\longrightarrow \mathbb{N}_0^n \\ (d_1, \dots, d_{n-1}) &\longrightarrow (d_1, \dots, d_{n-1}, c). \end{aligned}$$

Let $D \subset \mathbb{N}_0^n$, and naturally we define $\widehat{\text{proj}}(D) = \{\widehat{\text{proj}}(d) | d \in D\}$, and $\text{embed}_c(D') = \{\text{embed}_c(d) | d \in D'\}$ where $D' \subset \mathbb{N}_0^{n-1}$. In fact we can apply these mappings to any set $O \subset k^n$ or any matrix of n columns, because there is no danger of confusion. For example, let M be a matrix of n columns, and $\widehat{\text{proj}}(M)$ is a matrix of $n-1$ columns with the first $n-1$ columns of M reserved and the last one eliminated.

The embed_c mapping embeds an $n-1$ dimensional lower set into the n dimensional space. When the embed_c operation parameter c is zero, we can get an n dimensional lower set by mapping each element $d = (d_1, \dots, d_{n-1})$ to $d = (d_1, \dots, d_{n-1}, 0)$ as showed below.

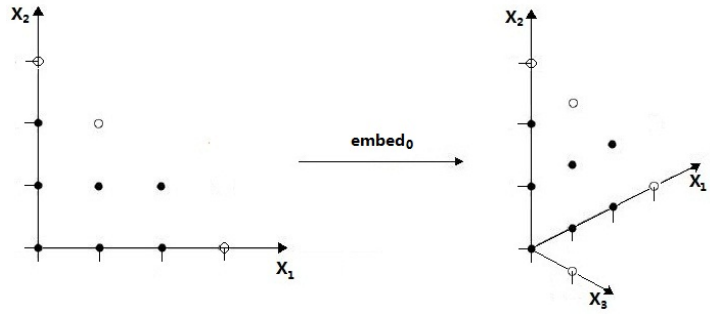


Fig. 6: Embed the lower set in 2-D space into 3-D space with parameter $c = 0$.

Blank circles represent the elements of the limiting sets. Note that after the embed_c mapping, there is one more blank circle. In this case, the limiting set is always increased by one element $(0, \dots, 0, 1)$.

In the case the embed_c operation parameter c is not zero, it is obvious that what we got is not a lower set any more. But there is another intuitive fact we should realize.

Theorem 1: D_0, D_1, \dots, D_k are $n-1$ dimensional lower sets, and $D_0 \supseteq D_1 \supseteq \dots \supseteq D_k$. Let $\hat{D}_i = \text{embed}_i(D_i)$, $i = 0, \dots, k$. Then $D = \bigcup_{i=0}^k \hat{D}_i$ is an n dimensional lower set, and $E(D) \subseteq C$ where $C = \bigcup_{i=0}^k \text{embed}_i(E(D_i)) \cup \{(0, \dots, 0, k+1)\}$.

Proof: First to prove D is a lower set. $\forall d \in D$, let $i = \text{proj}(d)$, then $d \in \hat{D}_i$ i.e. $\widehat{\text{proj}}(d) \in \widehat{\text{proj}}(\hat{D}_i) = D_i$. Because D_i is a lower set, hence for $j = 1, \dots, n-1$, if $d_j \neq 0$, then $\widehat{\text{proj}}(d) - \widehat{\text{proj}}(e_j) \in D_i$ where $e_j = (0, \dots, 0, 1, 0, \dots, 0)$ with the 1 situated at the j -th position. So $d - e_j \in \hat{D}_i \subseteq D$. For $j = n$, if $i = 0$, then we are finished. Else there must be $d - e_n \in \hat{D}_{i-1} \subseteq D$. Because if $d - e_n \notin \hat{D}_{i-1}$, we have $\widehat{\text{proj}}(d) \notin D_{i-1}$. Since we already have $\widehat{\text{proj}}(d) \in D_i$, this is contradictory to $D_i \subseteq D_{i-1}$.

Second, $\forall d \in E(D)$, $\widehat{\text{proj}}(d) \notin D_i$, $i = 0, \dots, k$. If $\widehat{\text{proj}}(d)$ is a zero tuple, then d_n must be $k+1$, that is $d \in C$. Else we know $d_n < k+1$. If $d_j \neq 0$, $j = 1, \dots, n-1$, then $d - e_j \in \text{embed}_{d_n}(D_{d_n})$. Then $\widehat{\text{proj}}(d) - \widehat{\text{proj}}(e_j) \in D_{d_n}$, that is $\widehat{\text{proj}}(d) \in E(D_{d_n})$. Finally with the embed_{d_n} operation we have $d \in \text{embed}_{d_n}(E(D_{d_n}))$ where $d_n < k+1$. So $d \in C$.

4. Addition of lower sets

In this section, we define the addition of lower sets which is the same with that in [2], the following paragraph and Fig.7 are basically excerpted from that paper with a little modification of expression.

To get a visual impression of what the addition of lower sets dose, look at the example in Fig.7. What is depicted there can generalize to arbitrary lower sets D_1 and D_2 in arbitrary dimension n and can be described as follows. Draw a coordinate system of \mathbb{N}_0^n and insert D_1 . Place a translate of D_2 somewhere on the X_2 -axis. The translate has to be sufficiently far out, so that D_1 and the translate D_2 do not intersect. Then take the elements of the translate of D_2 and drop them down along the X_2 -axis until they lie on top of the elements of D_1 . The resulting lower set is denoted by $D_1 + D_2$.

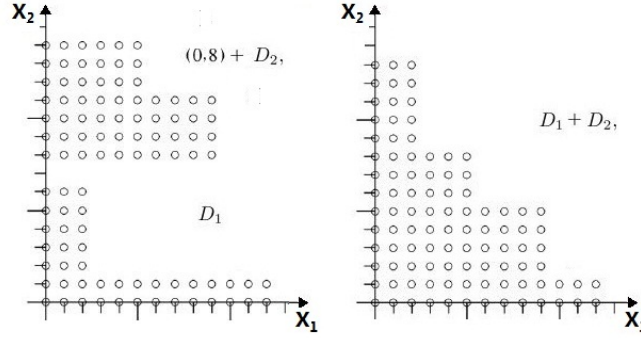


Fig. 7: Addition of D_1 and D_2 .

Intuitively, we define algorithm **AOL** to realize the addition of lower sets.

Algorithm AOL: Given two n dimensional lower sets D_1, D_2 , determine another lower set as the addition of D_1, D_2 , denoted by $D := D_1 + D_2$.

[step 1]: $D := D_1$;

[step 2]: If $\#D_2 = 0$ return D . Else pick $a \in D_2, D_2 := D_2 \setminus \{a\}$.

[step 2.1]: If $\#(D \cup \{a\}) = \#D$, add the last coordinate of a with 1. Go to [step 2.1]. Else $D := D \cup \{a\}$, go to [step 2].

Given n dimensional lower sets D_1, D_2, D_3 , the addition we defined satisfies:

- (a) $D_1 + D_2 = D_2 + D_1$,
- (b) $(D_1 + D_2) + D_3 = D_1 + (D_2 + D_3)$,
- (c) $D_1 + D_2$ is a lower set,
- (d) $\#(D_1 + D_2) = \#D_1 + \#D_2$.

These are all the same with that in [2]. And the proof can be referred to it.

As implied in the example of Section 2, when we want to get a polynomial with leading term d_3 showed in the right part of Fig.8, we need two polynomials with the leading terms d_1, d_2 which are not the elements of the lower sets and have the same degrees of X_1 as d_3 and the minimal degrees of X_2 as showed in the left part of Fig.8. In other words, $d_1 \notin D_1, d_2 \notin D_2, \widehat{proj}(d_1) = \widehat{proj}(d_2) = \widehat{proj}(d_3)$, $proj(d_1) + proj(d_2) = proj(d_3)$. It's easy to understand that these equations hold for the addition of three or even more lower sets.

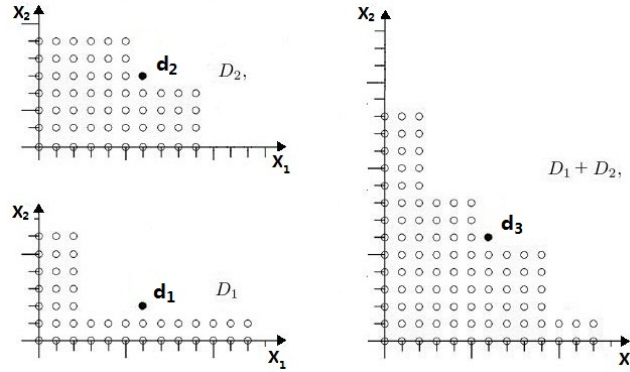


Fig.8: $\widehat{proj}(d_1) = \widehat{proj}(d_2) = \widehat{proj}(d_3)$, $proj(d_1) + proj(d_2) = proj(d_3)$.

We use algorithm **GLT** to get the leading terms d_1 and d_2 from d_3 respectively.

Algorithm GLT: Given $a \in \mathbb{N}_0^n$, and an n dimensional lower set D satisfying $a \notin D$. Determine another $r = (r_1, \dots, r_n) \in \mathbb{N}_0^n$ which satisfies that $r \notin D, \widehat{proj}(r) = \widehat{proj}(a)$ and $(r_1, \dots, r_{n-1}, r_n - 1) \in D$, denoted by $r := GLT(a, D)$.

[step 1]: Initialize r such as $\widehat{proj}(r) = \widehat{proj}(a)$ and $proj(r) = 0$.

[step 2]: if $r \notin D$, return r , else $r_n := r_n + 1$, go to [step 2].

Then $d_1 = GLT(d_3, D_1), d_2 = GLT(d_3, D_2)$.

Definition 3: For any $f \in k[X]$, view it as an element in $k(X_n)[X_1, \dots, X_{n-1}]$ and define $LC_n(f)$ to be the leading coefficient of f which is an univariate polynomial of X_n .

Algorithm GLP: D is an n dimensional lower set, $a \in \mathbb{N}_0^n$ and $a \notin D, G := \{f \in k[X]; \exists ed \in E(D), s.t. \text{ the leading term of } f \text{ is } ed\}$, algorithm **GLP** returns a polynomial p in the ideal $\langle G \rangle$ whose leading term is $GLT(a, D)$. Denoted by $p := GLP(a, D, G)$.

[step 1:] $c := GLT(a, D)$.

[step 2:] Select $c' \in E(D)$, s.t. c' is a factor of c . $d := \frac{c}{c'}$.

[step 3:] $p := f_{c'} \cdot d$ where $f_{c'}$ is an element of G whose leading term is c' .

Remark 1: $LC_n(f_{c'}) = LC_n(p)$ in [step 3]. Since c has the minimal degree of X_n according to algorithm **GLT**, there exists no element $c'' \in E(D)$ which is a factor of c satisfying $proj(c'') < proj(c)$. Hence monomial d in the algorithm does not conclude the variable X_n .

5. Associate a lower set $D(H)$ to a set of points H with multiplicity structures

For any given set of n dimensional points H with multiplicity structures, we can construct an n dimensional lower set $D(H)$ by induction.

Univariate case: $H = \{\langle p_1, D_1 \rangle, \dots, \langle p_t, D_t \rangle\}$, then the lower set is $D(H) = \{0, 1, \dots, \sum_{i=1}^t \#D_i\}$.

To pass from $n-1$ to n ($n \geq 2$), we first solve a **Special case**.

Special case: $H = \{\langle p_1, D_1 \rangle, \dots, \langle p_t, D_t \rangle\}$ is a set of n dimensional points with multiplicity structures where all the points share the same X_n coordinates. Write H in matrix form as $\langle \mathcal{P}, \mathcal{D} \rangle$ and all the entries in the last column of matrix \mathcal{P} have the same values. Classify the row vectors of $\langle \mathcal{P}, \mathcal{D} \rangle$ to get $\{\langle \mathcal{P}_0, \mathcal{D}_0 \rangle, \dots, \langle \mathcal{P}_w, \mathcal{D}_w \rangle\}$ according to the values of the entries in the last column of matrix \mathcal{D} and we guarantee the corresponding relationship between the row vectors of matrix \mathcal{P} and matrix \mathcal{D} holds in $\langle \mathcal{P}_i, \mathcal{D}_i \rangle$ ($0 \leq i \leq w$). All the entries in the last column of \mathcal{D}_i are the same i and the entries of the last column of \mathcal{P}_i stay the same too. Then eliminate the last columns of \mathcal{P}_i and \mathcal{D}_i to get $\langle \widehat{proj}(\mathcal{P}_i), \widehat{proj}(\mathcal{D}_i) \rangle$ which represents a set of $n-1$ dimensional points with multiplicity structures, by induction we get a lower set \hat{D}_i in $n-1$ dimensional space. Then we set

$$D(H) = \bigcup_{i=0}^w embed_i(\hat{D}_i).$$

Next we deal with the **General case**.

General case: $H = \{\langle p_1, D_1 \rangle, \dots, \langle p_t, D_t \rangle\}$ is a set of n dimensional points with multiplicity structures. Split the set of points: $H = H_1 \cup H_2 \cup \dots \cup H_s$. The points of H_i are in the same π -fibre, i.e. they have the same X_n coordinates c_i , $i = 1, \dots, s$, and $c_i \neq c_j, \forall i, j = 1, \dots, s, i \neq j$. According to the **Special case**, for each $i = 1, \dots, s$, we can get a lower set $D(H_i)$, then we set

$$D(H) = \sum_{i=1}^s D(H_i).$$

We now proof $D(H)$ is a lower set although it is easy to understand as long as the geometric interpretation involves. Since it is obviously true for **Univariate case**, induction over dimension would be helpful for the proof.

Proof: Assume $D(H)$ is a lower set for the $n-1$ dimensional situation and now we prove the conclusion for n dimensional situation ($n \geq 2$).

First to prove $D(H)$ of the **Special case** is a lower set.

We claim that $\langle \widehat{proj}(\mathcal{P}_i), \widehat{proj}(\mathcal{D}_i) \rangle$ represents an $n-1$ dimensional set of points with multiplicity structures ($i = 0, \dots, w$). For any $D \subseteq \mathbb{N}_0^n$, define $F_a(D) = \{d \in D \mid proj(d) = a\}$. Let $U = \{u \mid u \in \{1, \dots, t\}, F_i(D_u) \neq \emptyset\}$. So $\langle \widehat{proj}(\mathcal{P}_i), \widehat{proj}(\mathcal{D}_i) \rangle$ can be written in the form of $\{\langle \widehat{proj}(p_u), \widehat{proj}(F_i(D_u)) \rangle \mid u \in U\}$. Apparently $\widehat{proj}(F_i(D_u))$ is an $n-1$ dimensional lower set and can be viewed as the multiplicity structure of the point $\widehat{proj}(p_u)$. Hence $\langle \widehat{proj}(\mathcal{P}_i), \widehat{proj}(\mathcal{D}_i) \rangle$ is an $n-1$ dimensional set of points with multiplicity structures.

What's else, we assert $\widehat{proj}(\mathcal{P}_j)$ is a sub-matrix of $\widehat{proj}(\mathcal{P}_i)$, and $\widehat{proj}(\mathcal{D}_j)$ is a sub-matrix of $\widehat{proj}(\mathcal{D}_i)$, $0 \leq i < j \leq w$. Because of the corresponding relationship between the row vectors in \mathcal{P} and \mathcal{D} , we need only to prove $\widehat{proj}(\mathcal{D}_j)$ is a sub-matrix of $\widehat{proj}(\mathcal{D}_i)$. If it is not true, there exists a row vector g of $\widehat{proj}(\mathcal{D}_j)$ which is not a row vector of $\widehat{proj}(\mathcal{D}_i)$. That is, there exists b ($1 \leq b \leq t$) such that $embed_j(g)$ is an element of the lower set D_b , and $embed_i(g)$ is not included in any lower set D_a ($1 \leq a \leq t$). However since $i < j$ and $embed_j(g) \in D_b$, $embed_i(g)$ must be included in D_b . Hence our assertion is true.

Since $\widehat{proj}(\mathcal{P}_j)$ is a sub-matrix of $\widehat{proj}(\mathcal{P}_i)$, and $\widehat{proj}(\mathcal{D}_j)$ is a sub-matrix of $\widehat{proj}(\mathcal{D}_i)$, $0 \leq i < j \leq w$. According to the assumption of induction and the way we construct $D(H)$, we have $\hat{D}_i \supseteq \hat{D}_j$, $0 \leq i < j \leq w$, where \hat{D}_i, \hat{D}_j are both lower sets. Based on the **Theorem 1** in Section 3, $D(H) = \bigcup_{i=0}^w embed_i(\hat{D}_i)$ is a lower set, and $E(D(H)) \subseteq \bigcup_{i=0}^w embed_i(E(\hat{D}_i)) \cup \{(0, \dots, 0, w+1)\}$.

Then to prove $D(H)$ of **General case** is a lower set. Since $D(H_i), i = 1, \dots, s$ are lower sets, and the addition of lower sets is also a lower set according to Section 4, $D(H)$ is obviously a lower set. The proof is finished.

6. Associate a set of polynomials $\text{poly}(H)$ to $D(H)$

For every lower set constructed during the induction procedure showed in the last section, we associate a set of polynomials to it.

We begin with the univariate case as we did in the last section.

P-univariate case: $H = \{\langle p_1, D_1 \rangle, \dots, \langle p_t, D_t \rangle\}$, and $D(H) = \{0, 1, \dots, \sum_{i=1}^t \#D_i\}$. The set of polynomials associated to $D(H)$ is $\text{poly}(H) = \{\prod_{i=1}^t (X_1 - p_i)^{\#D_i}\}$.

Apparently, $\text{poly}(H)$ of **P-univariate case** satisfies the following **Assumption**.

Assumption: For any given $n - 1$ ($n > 1$) dimensional set of points H with multiplicity structures, there are the following conclusions. For any $\lambda \in E(D(H))$, there exists a polynomial $f_\lambda \in k[X]$ where $X = (X_1, \dots, X_{n-1})$ such that

- The leading term of f_λ under lexicographic ordering is X^λ .
- The exponents of all lower terms of f_λ lies in $D(H)$.
- f_λ vanishes on H .
- $\text{poly}(H) = \{f_\lambda | \lambda \in E(D(H))\}$.

When we construct the set of polynomials $\text{poly}(H)$, we should make sure the assumption always holds. Now let us consider the n ($n > 1$) dimensional situation and still begin with the special case.

P-Special case: Given a set of points with multiplicity structures $H = \{\langle p_1, D_1 \rangle, \dots, \langle p_t, D_t \rangle\}$ or in matrix form $\langle \mathcal{P} = (p_{ij})_{m \times n}, \mathcal{D} = (d_{ij})_{m \times n} \rangle$. All the given points have the same X_n coordinates, i.e. the entries in the last column of \mathcal{P} are the same. We compute $\text{poly}(H)$ following the steps below.

[step 1]: $c := p_{1n}$; $w := \max\{d_{in}; i = 1, \dots, m\}$.

[step 2]: $\forall i = 0, \dots, w$, define \mathcal{SD}_i as a sub-matrix of \mathcal{D} containing all the row vectors whose last coordinates equal to i . Extract the corresponding row vectors of \mathcal{P} to form matrix \mathcal{SP}_i , and the corresponding relationship between the row vectors in \mathcal{P} and \mathcal{D} holds for \mathcal{SP}_i and \mathcal{SD}_i .

[step 3]: $\forall i = 0, \dots, w$, eliminate the last columns of \mathcal{SP}_i and \mathcal{SD}_i to get $\langle \tilde{\mathcal{SP}}_i, \tilde{\mathcal{SD}}_i \rangle$ which represents a set of points in $n - 1$ dimensional space with multiplicity structures. According to the induction assumption, we have the polynomial set $\tilde{G}_i = \text{poly}(\langle \tilde{\mathcal{SP}}_i, \tilde{\mathcal{SD}}_i \rangle)$ associated to the lower set $\tilde{D}_i = D(\langle \tilde{\mathcal{SP}}_i, \tilde{\mathcal{SD}}_i \rangle)$.

[step 4]: $D := \bigcup_{i=0}^w \text{embed}_i(\tilde{D}_i)$. Multiply every element of \tilde{G}_i with $(X_n - c)^i$ to get G_i . $\tilde{G} := \bigcup_{i=0}^w G_i \cup \{(X_n - c)^{w+1}\}$.

[step 5]: Eliminate the polynomials in G whose leading term is not included in $E(D)$ to get $\text{poly}(H)$.

Theorem 2: The $\text{poly}(H)$ got in **P-Special case** satisfies the **Assumption**.

Proof: According to the Section 5, $\langle \tilde{\mathcal{SP}}_i, \tilde{\mathcal{SD}}_i \rangle$ represents an $n - 1$ dimensional set of points with multiplicity structures for $i = 0, \dots, w$. And $\tilde{D}_j \supseteq \tilde{D}_i, 0 \leq j \leq i \leq w$. D is a lower set and $E(D) \subseteq \bigcup_{i=0}^w \text{embed}_i(E(\tilde{D}_i)) \cup \{(0, \dots, 0, w + 1)\}$.

For $\lambda = (0, \dots, 0, w + 1) \in E(D)$, we have $f_\lambda = (X_n - c)^{w+1}$. It is easy to check that it satisfies the first three terms of the **Assumption**.

For any other element ed of $E(D)$, $\exists k$ s.t. $ed \in \text{embed}_k E(\tilde{D}_k)$. So let \tilde{ed} be the element in $E(\tilde{D}_k)$ such that $ed = \text{embed}_k(\tilde{ed})$. We have $f_{\tilde{ed}}$ vanishes on $\langle \tilde{\mathcal{SP}}_k, \tilde{\mathcal{SD}}_k \rangle$ whose leading term is $\tilde{ed} \in E(\tilde{D}_k)$ and the lower terms belong to \tilde{D}_k . According to the algorithm $f_{ed} = (X_n - c)^k \cdot f_{\tilde{ed}} \in \text{poly}(H)$.

First it is easy to check that the leading term of f_{ed} is ed since $ed = \text{embed}_k(\tilde{ed})$.

Second, the lower terms of f_{ed} are all in the set $S = \bigcup_{j=0}^k \text{embed}_j(\tilde{D}_k)$ because all the lower terms of $f_{\tilde{ed}}$ are in the set \tilde{D}_k . $\tilde{D}_0 \supseteq \tilde{D}_1 \supseteq \dots \supseteq \tilde{D}_k$, so $\text{embed}_j(\tilde{D}_k) \subset \text{embed}_j(\tilde{D}_j)$ ($0 \leq j \leq k$), hence $S \subseteq D = \bigcup_{j=0}^w \text{embed}_j(\tilde{D}_j)$ and the second term of the **Assumption** is satisfied.

Third, we are going to prove that f_{ed} vanishes on all the functionals defined by $\langle \mathcal{P}, \mathcal{D} \rangle$, i.e. all the functionals defined by $\langle \mathcal{SP}_i, \mathcal{SD}_i \rangle$ ($i = 0, \dots, w$).

When $i \neq k$, we write all the functionals defined by $\langle \mathcal{SP}_i, \mathcal{SD}_i \rangle$ in this form: $L' \cdot \frac{\partial^i}{\partial X_n^i} |_{X_n=c}$ where L' is an $n - 1$ variable functional. Since $f_{ed} = (X_n - c)^k \cdot f_{\tilde{ed}}$, apparently f_{ed} vanishes on these functionals.

For $i = k$, denote by L the functionals defined by $\langle \tilde{\mathcal{SP}}_i, \tilde{\mathcal{SD}}_i \rangle$, and $f_{\tilde{ed}}$ vanishes on L . All the functionals defined by $\langle \mathcal{SP}_k, \mathcal{SD}_k \rangle$ can be written in this form: $L'' \cdot \frac{\partial^k}{\partial X_n^k} |_{X_n=c}$ where $L'' \in L$. Since $f_{ed} = (X_n - c)^k \cdot f_{\tilde{ed}}$, apparently f_{ed} vanishes on these functionals.

So f_{ed} vanish on H , and f_{ed} satisfies the first three terms of the **Assumption**.

In summary $\text{poly}(H)$ satisfies the **Assumption**, and we finish the proof.

Remark 2: For $f_\lambda \in \text{poly}(H)$, $\lambda \in E(D)$ where $\text{poly}(H)$ is the result got in the algorithm above, we have the conclusion that $LC_n(f_\lambda) = (X_n - c)^{\text{proj}(\lambda)}$.

P-General case: Given a set of points with multiplicity structures H or in matrix form $\langle \mathcal{P} = (p_{ij})_{m \times n}, \mathcal{D} = (d_{ij})_{m \times n} \rangle$, we are going to get $\text{poly}(H)$.

[step 1]: Write H as $H = H_1 \cup H_2 \cup \dots \cup H_s$ where H_i ($1 \leq i \leq s$) is a π -fibre ($\pi : H \mapsto k$ such that $\langle p = (p_1, \dots, p_n), D \rangle \in H$ is mapped to $p_n \in k$) i.e. the points of H_i have the same X_n coordinates c_i , $i = 1, \dots, s$, and $c_i \neq c_j, \forall i, j = 1, \dots, s, i \neq j$.

[step 2]: According to the **P-Special case**, we have $D'_i = D(H_i), G_i = \text{poly}(H_i)$. Write H_i as $\langle \mathcal{P}_i, \mathcal{D}_i \rangle$, and define w_i as the maximum value of the elements in the last column of \mathcal{D}_i .

[step 3]: $D := D'_1, G := G_1, i := 2$.

[step 4]: If $i > s$, go to [step 5]. Else

[step 4.1]: $D := D + D'_i; \hat{G} := \emptyset$. View $E(D)$ as a monomial set $MS := E(D)$.

[step 4.2]: If $\#MS = 0$, go to [step 4.7], else select the minimal element of MS under lexicographic ordering, denoted by LT . $MS := MS \setminus \{LT\}$.

[step 4.3]:

$$f_1 := GLP(LT, D, G), f_2 := GLP(LT, D'_i, G_i).$$

$$v_k := \text{proj}(g_k), \text{ where } g_k := GLT(LT, D'_k), k = 1, \dots, i.$$

[step 4.4]:

$$q_1 := f_1 \cdot (X_n - c_i)^{w_i+1}; \quad q_2 := f_2 \cdot \prod_{k=1}^{i-1} (X_n - c_k)^{w_k+1}.$$

$$pp1 := (X_n - c_i)^{w_i+1-v_i}; \quad pp2 := \prod_{k=1}^{i-1} (X_n - c_k)^{w_k+1-v_k}.$$

[step 4.5]: Use Extended Euclidean Algorithm to compute r_1 and r_2 s.t. $r_1 \cdot pp1 + r_2 \cdot pp2 = 1$.

[step 4.6]: $f := r_1 \cdot q_1 + r_2 \cdot q_2$. Reduce f with the elements in \hat{G} to get f' ; $\hat{G} := \hat{G} \cup \{f'\}$. Go to [step 4.2].

[step 4.7]: $G := \hat{G}$. $i := i + 1$. Go to [step 4].

[step 5]: $\text{poly}(H) := G$.

Theorem 3: The $\text{poly}(H)$ got in **p-General case** satisfies the **Assumption**.

proof: We need only to prove the situation that $s \geq 2$ in [step 1].

For $i = 2$, $D = D'_1 + D'_2$. $\forall ed \in E(D)$, $v := \text{proj}(ed)$ and $X_0 := \frac{X^{ed}}{X_n^v}$. According to Section 4, we have $v = v_1 + v_2$. Based on the **Remark 1** and **Remark 2**, f_1 and f_2 can be written as polynomials of $k(X_n)[X_1, \dots, X_{n-1}]$: $f_1 = X_0 \cdot (X_n - c_1)^{v_1} + \text{the rest}$ and $f_2 = X_0 \cdot (X_n - c_2)^{v_2} + \text{the rest}$ and none of the monomials in *the rest* is greater than or equal to X_0 . Because f_1 and $(X_n - c_1)^{w_1+1}$ vanish on H_1 , f_2 and $(X_n - c_2)^{w_2+1}$ vanish on H_2 , we know that $q_1 = f_1 \cdot (X_n - c_2)^{w_2+1}$ and $q_2 = f_2 \cdot (X_n - c_1)^{w_1+1}$ both vanish on $H_1 \cup H_2$. Then f vanishes on $H_1 \cup H_2$ where $f = r_1 \cdot q_1 + r_2 \cdot q_2$.

$$\begin{aligned} f &= X_0 \cdot (X_n - c_1)^{v_1} \cdot (X_n - c_2)^{v_2} (r_1 \cdot (X_n - c_2)^{w_2+1-v_2} + r_2 \cdot (X_n - c_1)^{w_1+1-v_1}) + \text{the rest} \\ &= X_0 \cdot (X_n - c_1)^{v_1} \cdot (X_n - c_2)^{v_2} (r_1 \cdot pp1 + r_2 \cdot pp2) + \text{the rest} \\ &= X_0 \cdot (X_n - c_1)^{v_1} \cdot (X_n - c_2)^{v_2} + \text{the rest} \end{aligned}$$

None monomial in *the rest* is greater than or equal to X_0 , so the leading term of f is apparently $X_0 \cdot X_n^v$ which is equal to ed . Moreover we naturally have the following **Proposition 1** for $i = 2$.

Proposition 1: For every polynomial f we get in the algorithm, $LC_n(f) = \prod_{j=1}^i (X_n - c_j)^{v_j}$.

When $i > 2$, assume the **Proposition 1** holds for $i - 1$. $\forall ed \in E(D)$, $v := \text{proj}(ed)$ and $X_0 := \frac{X^{ed}}{X_n^v}$. According to Section 4, we have $v = v_1 + \dots + v_i$. Based on the **Proposition 1**, **Remark 1** and **Remark 2**, f_1 and f_2 can be written as polynomials of $k(X_n)[X_1, \dots, X_{n-1}]$: $f_1 = X_0 \cdot \prod_{j=1}^{i-1} (X_n - c_j)^{v_j} + \text{the rest}$ and $f_2 = X_0 \cdot (X_n - c_i)^{v_i} + \text{the rest}$ and none of the monomials in *the rest* is greater than or equal to X_0 . Because f_1 and $\prod_{j=1}^{i-1} (X_n - c_j)^{w_j+1}$ vanish on $\bigcup_{j=1}^{i-1} H_j$, f_2 and $(X_n - c_i)^{w_i+1}$ vanish on H_i , we know that $q_1 = f_1 \cdot (X_n - c_i)^{w_i+1}$ and $q_2 = f_2 \cdot \prod_{j=1}^{i-1} (X_n - c_j)^{w_j+1}$ both vanish on $\bigcup_{j=1}^i H_j$. Then f vanishes on $\bigcup_{j=1}^i H_j$ where $f = r_1 \cdot q_1 + r_2 \cdot q_2$.

$$\begin{aligned} f &= X_0 \cdot \prod_{j=1}^i (X_n - c_j)^{v_j} (r_1 \cdot (X_n - c_i)^{w_i+1-v_i} + r_2 \cdot \prod_{j=1}^{i-1} (X_n - c_j)^{w_j+1-v_j}) + \text{the rest} \\ &= X_0 \cdot \prod_{j=1}^i (X_n - c_j)^{v_j} (r_1 \cdot pp1 + r_2 \cdot pp2) + \text{the rest} \end{aligned}$$

$$= X_0 \cdot \prod_{j=1}^i (X_n - c_j)^{v_j} + \text{the rest}$$

None monomial in *the rest* is greater than or equal to X_0 and the leading term of f is apparently $X_0 \cdot X_n^v$ which is equal to ed . Hence the **Proposition 1** holds for arbitrary i .

Therefore we have proved that for any element $ed \in E(D)$, $f_{ed} := f$ vanishes on H and the leading term is ed . In the algorithm, we compute f_{ed} in turn according to the lexicographic ordering of the elements of $E(D)$. Once we get a polynomial, we use the polynomials we got previously to reduce it ([step 4.6]). Now to prove the lower terms of f' are all in D after such a reduction operation.

Let D be a lower set, a be a monomial, define $L(a, D) = \{b \in \mathbb{N}_0^n; b \prec a, b \in D\}$. Given any $d \notin D$, there exist only two situations: $d \in E(D)$ or $d \notin E(D)$ but $\exists d' \in E(D)$, s.t. d' is a factor of d . Of course $d' \prec d$.

The very first vanishing polynomial we got in the algorithm is an univariate polynomial of X_n with leading term being T . It is easy to check it's lower terms are in D . Since the polynomial is a vanishing polynomial, we can say that T can be represented as the linear combination of the elements of $L(T, D)$.

Since T is the first element which is not in D under lexicographic ordering. We **assume** that there exists such a monomial $M \notin D (M \succ T)$ that $\forall m \prec M (m \notin D)$, m can be represented as the linear combination of the elements of $L(m, D)$. Now to prove M could be represented as the linear combination of the elements of $L(M, D)$.

If $M \in E(D)$, then the algorithm provides us a vanishing polynomial whose leading term is M i.e. that M can be represented as the combination of the terms which are all smaller than M . According to the assumption, for any lower term m ($m \notin D$) of the polynomial, m can be represented as the linear combination of the elements of $L(m, D)$, then M could be represented as the linear combination of the elements of $L(M, D)$.

If $M \notin E(D)$, there exists $d' \in E(D)$ s.t. $M = M' \cdot d'$. Since $d' \prec M$, according to the assumption, we can substitute d' with the linear combination of the elements of $L(d', D)$. Since all the elements in $L(d', D)$ are smaller than d' , then M could be represented as the combination of elements which are all smaller than M . Then for the same reason described in the last paragraph, M could be represented as the linear combination of the elements of $L(M, D)$.

Therefore specially for any $ed \in E(D)$, all the lower terms of the polynomial f_{ed} we got in the algorithm after the reduce operation are in D , and the proof is done.

Theorem 4: Given a set of points H with multiplicity structures, $\text{poly}(H)$ is the reduced Gröbner basis of the vanishing ideal $I(H)$ and $D(H)$ is the quotient basis under lexicographic ordering.

Proof: Let m be the number of functionals defined by H and then $m = \dim(k[X]/I(H))$. Denote by J the ideal generated by $\text{poly}(H)$. According to the **Assumption**, $\text{poly}(H) \subseteq I(H)$. So $\dim(k[X]/I(H)) \leq \dim(k[X]/J)$. Let C be the leading terms of polynomials in J under lexicographic ordering, then $C \supseteq \bigcup_{\beta \in E(D(H))} (\beta + \mathbb{N}_0^n)$ where the latter union is equal to $\mathbb{N}_0^n - D(H)$. Then we can get $C' = \mathbb{N}_0^n - C \subseteq D(H)$. Because $k[X]/J$ is isomorphic as a k -vector space to the k -span of C' , here C' is viewed as a monomial set. So $\dim(k[X]/J) \leq \#D(H) = m$. Hence we have

$$m = \dim(k[X]/I(H)) \leq \dim(k[X]/J) \leq m.$$

Therefore $J = I(H)$, where $J = \langle \text{poly}(H) \rangle$. Hence apparently $\text{poly}(H)$ is exactly the reduced Gröbner basis of the vanishing ideal under lexicographic ordering, and $D(H)$ is the quotient basis.

7. Intersection of ideals and some applications

Some steps of our algorithm actually do the work of computing the intersection of two ideals, but we note that the information of the zeros of the ideals is necessary there (see [step 4.1] - [step 4.7] of **p-General case** in Section 6). We now bring up a new algorithm to compute the intersection of two ideals which does not require the information of the zeros of the ideals.

Lemma 1: G is the reduced Gröbner basis of some n -variable polynomial ideal under lexicographic ordering with $X_1 \succ X_2 \succ \dots \succ X_n$. Define $p_0(G)$ as the univariate polynomial of X_n in G . View $g \in G$ as polynomial of $K(X_n)[X_1, \dots, X_{n-1}]$ and define $LC_n(g)$ to be the leading coefficient of g which is an univariate polynomial of X_n and we have the conclusion that $LC_n(g)$ is always a factor of $p_0(G)$.

Proof: In fact **Proposition 1** in Section 6 holds for any given reduced Gröbner basis under lexicographic ordering since it is unique and can be constructed in the way our algorithm offers. According to the proposition, $\forall f \in G$, $LC_n(g) = \prod_{j=1}^s (X_n - c_j)^{v_j}$ and $v_j \leq w_j + 1$. $p_0(G) = \prod_{j=1}^s (X_n - c_j)^{w_j + 1}$. Hence the proof is done.

Based on **Proposition 1** and **Lemma 1**, we give the algorithm **Intersection** to compute the intersection of two ideals I_1 and I_2 which are represented by the lexicographic ordering reduced Gröbner

bases G_1 and G_2 and the greatest common divisor of $p_0(G_1)$ and $p_0(G_2)$ equals to 1. Denote by $Q(G)$ the quotient basis where G is the reduced Gröbner basis. Algorithm **GP** is a sub-algorithm called in algorithm **Intersection**.

Algorithm **GP**: G is a reduced Gröbner basis, for any given monomial LT which is not in $Q(G)$, we get a polynomial p in $\langle G \rangle$ whose leading term is a factor of LT : the X_1, \dots, X_{n-1} components of the leading term are the same with that of LT and the X_n component has the lowest degree. Denoted by $p := GP(LT, G)$.

[step 1:] $G' := \{g \in G \mid \text{the leading monomial of } g \text{ is a factor of } LT\}$.

[step 2:] $G'' := \{g \in G' \mid \nexists g' \in G', \text{ s.t. the degree of } X_n \text{ of the leading monomial of } g' \text{ is lower than that of } g\}$.

[step 3:] Select one element of G'' and multiply it by a monomial of X_1, \dots, X_{n-1} to get p whose leading monomial is LT .

Algorithm **Intersection**: G_1 and G_2 are the reduced Gröbner bases of two different ideals satisfying that $GCD(p_0(G_1), p_0(G_2)) = 1$. Return the reduced Gröbner basis of the intersection of these two ideals, denoted by $G := Intersection(G_1, G_2)$.

[step 1:] $D := Q(G_1) + Q(G_2)$. View $E(D)$ as a monomial set. $G := \emptyset$.

[step 2:] If $E(D) = \emptyset$, the algorithm is done. Else select the minimal element of $E(D)$, denoted by T . $E(D) := E(D) \setminus \{T\}$.

[step 3:]

$$f_1 := GP(T, G_1), \quad f_2 := GP(T, G_2).$$

$$q_1 := f_1 \cdot p_2, \quad q_2 := f_2 \cdot p_1.$$

[step 4:]

$$t_1 := \frac{p_0(G_2)}{LC_n(f_2)}, \quad t_2 := \frac{p_0(G_1)}{LC_n(f_1)}.$$

[step 5:] Use Extended Euclidean Algorithm to find r_1, r_2 s.t.

$$r_1 \cdot t_1 + r_2 \cdot t_2 = 1.$$

[step 6:] $f := q_1 \cdot r_1 + q_2 \cdot r_2$. Reduce f with G to get f' , and $G := G \cup \{f'\}$. Go to [Step 2].

Because the algorithm is essentially the same with [step 4.1] - [step 4.7] of **p-General case** in Section 6, here we don't give the proof.

The **Proposition 1** and **Lemma 1** reveal important property of the reduced Gröbner basis under lexicographic ordering. If a set of polynomials does not have this property, it is surely not a reduced Gröbner basis.

It is well-known that the Gröbner basis of an ideal under lexicographic ordering holds good algebraic structures and hence is convenient to use for polynomial system solving. To compute the zeros of an zero dimensional ideal with the reduced Gröbner basis G , we need first compute the roots of $p_0(G)$. Since $LC_n(g)$ ($g \neq p_0(G)$, $g \in G$) is a factor of $p_0(G)$, compute the roots of $LC_n(g)$ which has a smaller degree would be helpful for saving the computation cost.

8. Conclusion

Based on the algorithm **Intersection** in Section 7, the algorithm of **p-General case** in Section 6 can be simplified. The last sentence in [step 2] can be deleted and we can replace [step 4.3] and [step 4.4] by:

[step 4.3]:

$$f_1 := GLP(LT, D, G), \quad f_2 := GLP(LT, D'_i, G_i).$$

[step 4.4]:

$$q_1 := f_1 \cdot p_0(G_i); \quad q_2 := f_2 \cdot p_0(G).$$

$$pp1 := \frac{p_0(G_i)}{LC_n(f_2)}; \quad pp2 := \frac{p_0(G)}{LC_n(f_1)}.$$

During the induction of the algorithm in Section 6, we can record the leading coefficients for later use to save the computation cost and the computation cost is mainly on the Extended Euclid Algorithm. However the advantage of our algorithm is not fast computation, after all it depends on how many times we need to use the Extended Euclid Algorithm.

Our algorithm has an explicit geometric interpretation which reveals the essential connection between the relative position of the points with multiplicity structures and the quotient basis of the vanishing ideal.

The algorithm offers us a new perspective of view to look into the reduced Gröbner basis which can help us understand the problem better. **Lemma 1** and the algorithm to compute the intersection of two ideals are the direct byproducts of our algorithm.

Since we finished the paper [1] previously which gives an algorithm to get the minimal monomial basis of Birkhoff interpolation problem with little computation cost, we have always believed that the algorithm could be interpreted in a more geometric way and the proof should be more beautiful and much easier to understand. The proof in [1] is so complicated that we ourselves don't like it. And it would be great if we can get the interpolation polynomial with little computation cost instead solving the linear equations since the minimal monomial basis can already be got in a simple way. That's why we began to study the vanishing ideal of the set of points with multiplicity structures which is essentially a special case of Birkhoff interpolation problem.

I still remember the moment when I first read the paper [2] written by Mathias Lederer in which the quotient basis and Gröbner basis can be got in a geometric way. I told myself that this was just what we wanted. Paper [2] concentrates on the the vanishing ideal of the set of points with no multiplicity structures in affine space. Although whether or not the points are with multiplicity structures matters much, the paper really inspired us a lot. Our algorithm also uses induction over variables and the definition of addition of lower sets is essentially the same with that in paper [2]. However during the induction procedure, we have to consider **p-Special case** and **p-General case**. This consideration, on one hand, clearly indicates the geometric meaning of the multiplicity structures of points, on the other hand, means a lot for our capacity of applying the algorithm of intersection of two ideals. In paper [2], the author uses Lagrange interpolation method to get the vanishing polynomial of all points from the polynomials vanishing on subsets of the points. However the Lagrange interpolation method could just not work for our problem because the points are with multiplicity structures. In this paper, we creatively use the Extended Euclidean Algorithm. Thanks goes to paper [2] and the author, after we solved the problem of the vanishing ideal of the set of points with multiplicity structures, we will move on to the Birkhoff problem.

9. References

- [1]Na Lei, Junjie Chai, Peng Xia, Ying Li, A fast algorithm for multivariate Birkhoff interpolation problem, *Journal of Computational and Applied Mathematics* 236 (2011) 1656-1666.
- [2]Mathias Lederer, The vanishing ideal of a finite set of closed points in affine space, *Journal of Pure and Applied Algebra* 212 (2008) 1116-1133.
- [3]Hans J.Stetter. *Numerical Polynomial Algebra*. Chapter 2. SIAM, Philadelphia, PA, USA. 2004.
- [4]M.G. Marinari, H.M. Möller, T. Mora, Gröbner bases of ideals defined by functionals with an application to ideals of projective points, *J. AAECC* 4 (2) (1993) 103-145.
- [5]Bálint Felszeghy, Balázs Ráth, Lajos Ronyai, The lex game and some applications, *Journal of Symbolic Computation* 41 (2006) 663-681.
- [6]L. Cerlinco, M. Mureddu, From algebraic sets to monomial linear bases by means of combinatorial algorithms, *Discrete Math* 139 (1995) 73-87.